Interex HP3000 Madrid Conference, March 1986
Theme: 'Migration to 2001'

. . . a picture's worth a thousand words . . . and on a computer can take more processing and storage than 10,000 words . . . in the Migration to 2001, if you find your graphics needs growing, here are some tips on quality and performance that may help you . . . .

A Blend of HP3000 and HP9000 for Computer Graphics

Sam Boles, Member Technical Staff

HEWLETT PACKARD

We all know that a picture's worth a thousand words.

And those of us who've drawn pictures with a computer know a picture can take more processing and storage than 10,000 words.

Then once you've gotten the basics under your belt, you get harder to please. You want resolution. You want performance. You want those end-points to meet, you want your curves smoother . . . you might even want animation . . . but, for sure, you don't want to wait.

This growing colony of computer artists with their growing appetite for the artistic can bring a multi-user commercial machine to its knees . . . unless you resort to second-order Distributed Systems.

You've experienced first-order DS with your HP3000 talking to others like it, sharing programs and databases. There's a second-order distribution that networks into your HP3000 the high-performance high-resolution graphics capabilities of the HP9000 for data-words-graphics integration on your HP3000 with its laser printers and other powerful peripherals.

This description of graphics and technical publication techniques may give you some productivity ideas for your own installation.

. . . why graphics? . . . for those who can't read . . .

Once upon a time, there was a magnificent piece of ***americana*** called ***Life***. It came out once a week. And it cost only a dime. It was the week that was, in pictures. The glory of victory, the agony of defeat. The blood, sweat and tears that started wars and finished wars. The laughter, the sobs, the bad, beautiful, noble and ludicrous of the human condition -- photographed by some of the most courageous men and women in the history of journalism.

But not everyone viewed this piece of ***americana*** the same way. A young undergraduate (an English major) once remarked,

"Life is for those who can't read . . .

Time, for those who can't think . . . ."

Without commenting on its validity let's see if we can leverage this wisdom of a generation past, and come up with an answer to the question

Why Graphics?

Graphics is for those who can't read. No, it's not that they can't read because they can't read. They can't read because ***they don't have time to read.***

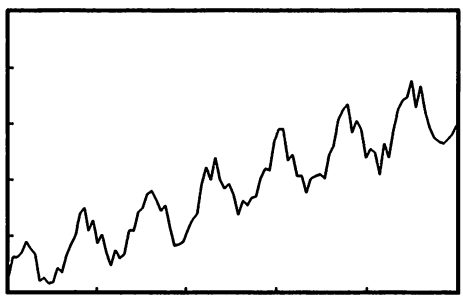
Let's look at an example out of one of the HP9000 reference manuals. Read these numbers:

```
0.1610 0.1625 0.1625 0.1628 0.1636
0.1631 0.1627 0.1608 0.1610 0.1606
0.1607 0.1617 0.1614 0.1626 0.1634
0.1640 0.1656 0.1660 0.1644 0.1651
0.1635 0.1641 0.1628 0.1619 0.1630
0.1624 0.1627 0.1644 0.1644 0.1657
0.1660 0.1670 0.1672 0.1666 0.1658
0.1662 0.1646 0.1633 0.1634 0.1636
0.1645 0.1652 0.1656 0.1677 0.1689
0.1680 0.1696 0.1680 0.1674 0.1677
0.1669 0.1655 0.1665 0.1662 0.1667
0.1668 0.1681 0.1688 0.1687 0.1707
0.1716 0.1716 0.1694 0.1698 0.1683
0.1683 0.1671 0.1681 0.1683 0.1684
0.1681 0.1698 0.1705 0.1723 0.1730
0.1734 0.1714 0.1722 0.1716 0.1696
0.1702 0.1699 0.1684 0.1706 0.1696
0.1715 0.1730 0.1737 0.1739 0.1751
0.1732 0.1747 0.1729 0.1717 0.1710
0.1707 0.1706 0.1709 0.1713 0.1720
```

Did you read them? No, of course you didn't. You ***don't have time*** to read . . . ***really*** read . . . a hundred numbers. And if you did have time you wouldn't waste it like that. (Notice how boring it gets after about the third digit?)

Besides why read a hundred numbers when today's technology can read them for you and maybe tell you something you might have missed -- because it's in ***between the lines.*** Just browsing probably gives you the overall trend, but how about periodic motions and number of cycles?

Why Graphics?
The 100 Numbers You Didn't Read



Better, Easier, FASTER!

. . . and for those who can't think . . . again, because they don't have the time . . .

This maybe tells us something else, too: Graphics is ***for those who can't think***. No, it's not that they can't think because they can't think. They can't think because ***they don't have time to think.***

Sure, they could take a pencil and paper and calculate the deltas and get the pattern clusters in a few minutes. But that few minutes has an ***opportunity cost***. What they ***could've done.*** Like get the corrective action launched. Or the next step of the design underway. Or whatever is the ***real work*** they could've been doing.

. . . to save time . . . the Ultimate Unreplenishable . . . at a rate of a thousand words per . . .

It's the old story of "a picture's worth a thousand words." You get the message to your audience ***better*** . . . ***faster***. ***You*** save time. ***They*** save time.

Let's look at this thing called ***time***. It's a unique commodity. Or maybe it's ***not a commodity*** -- ever try to buy some time? Anyway, it's ***unique***.

Remember back in the early 70's when we had to stand in line to get gas? Geologists for years had been telling us we were burning oil ***faster than it was being replenished***. Then we finally got the message: That meant we could run out. So the oil owners lowered production and raised the price. And suddenly there we were, waiting in line for stuff that not long before that a Gulf station in Los Angeles would sell for 18.9 cents a gallon during a gas war . . . and clean your windshield while you bought it.

All of that for something ***relatively*** unreplenishable. But look at time. The ***Ultimate*** Unreplenishable.

If oil is ***slow*** at replenishing, it's nothing compared to Time. Time ***doesn't replenish at all***. You burn it and it's gone. Forever. And we burn it every second of every minute of every day. Not just when we drive to work. And we burn more of it faster all the time. Ask anyone who's been around awhile about how much faster you burn it as years go by. And how much ***more*** of it you burn.

The ***Ultimate Unreplenishable***.

Anything that improves performance in time utilization deserves attention. And graphics is one of those things.

So much for *Why Graphics?* Let's look at the evolution of the computer graphics artist.

. . . the evolution of the artist . . . basics under your belt, you get harder to please . . .

Remember a few years ago when ***DSG*** (Decision Support Graphics) brought charts right to our terminal on the HP3000? Bar charts, line charts, pie charts. We could eavesdrop a plotter on the line and get hard copy on the spot in minutes. Then there was ***HPDRAW*** to build the text and picture slides to go with the charts.

We could build our slide presentations quickly and conveniently. And update with the latest numbers in a matter of minutes.

The magic of ***Interface Reduction***.

We no longer had to queue up at the graphics department. And wait for the typesetter. We ***reduced*** these ***interfaces*** to zero. We saved time and money. A quantum leap in ***Productivity thru Interface Reduction***.

It was great.

Then as the elation wore off we noticed you could tell the computer slides from the typeset slides. It was the letters. Those stick letters. Like tapioca: ***good, but not exciting***.

We needed better letters. Nice smooth spline curves, with fill and boundary in different colors. And a wider range of fonts and sizes.

So DRAW II arrived with really world-class letters. A quantum leap in ***professional quality lettering***.

It was great.

Then as the elation wore off we noticed that next to the beautiful spline letters you really noticed when the end points in our drawings

didn't quite meet, and some of the detail was a little ragged.

We needed ***better resolution***, and better and faster zoom, pan, grid snapping and . . .

On top of all that, something was happening in our work area.

Our colleagues were trying to figure out how we were able to get such good presentations ***so fast*** and ***still come in under budget***. They saw we were using DSG and DRAW, and they started to do the same thing.

CPU utilization began to grow. The last 3 days before quarterly review, ***response time*** got really ***slow***. We upgraded to a 68, and that helped. But not enough to keep up with the growing popularity of the tool.

. . . the issues of resolution, response time and load balancing . . .

As we experienced more and more positive results from our graphics, we wanted more and more quantity and quality.

More people were starting to use the tool. And the positive results from using the tool made them use it more.

Each iteration made the users more proficient with the tool, enabling them to do a better job of the next slide and increasing their appetite for perfection proportionately.

The ***need for fast-response high-resolution graphics tools evolved as the degradation of response time evolved***, aggravating the economic imbalance with a diminishing supply of CPU cycles, disc I/Os and main memory being confronted with an increasing demand for a higher service level by more users.

Compound the situation with a ***25 per cent annual growth rate*** and you get the scene that led us into second-order Distributed Systems for our computer graphics.

We already had our 3000's DS-linked so we could get at programs and databases on neighboring machines. And if we had a heavy-duty crunch that we needed to run we could off-load this to one of the light-load nodes.

But with the graphics overload we were looking for ***more than just CPU cycles***. We were looking for functionality. A richer command set. A more natural human interface. Higher resolution. Faster graphics performance in first-draws, redraws, transformation of primitives and cells.

We turned to our CAD-CAM family, the HP9000, and found what we were looking for.

Let's look at where the 9000 was coming from.

. . . the HP9000 genealogy . . . a "PC" before they invented the word . . .

If you trace the roots of today's HP9000 family, you go back to the days when we used words like "programmable calculator" and "desktop computer." If you look at the HP9825 (***circa*** mid-70's) you see the ***low-cost, small footprint, portable and individual work station*** that might've been called a "PC" if we'd used that kind of language in the medieval days of 10 years ago.

When we retired its jersey some time back, the HP9825 had been one of the top unit sellers in the history of Hewlett-Packard.

The tradition evolved into the HP9000/200 and /300 with 8-16 Megahertz processors with 3- and 4-plane color graphics, and the 3-CPU HP9000/500. The 300 and 500 support 8 planes of color (that's 256 colors from a palette of 16 million) with a graphics accelerator that pumps ***60,000 vectors a second over a 2 Megabyte bandwidth bus***.

What all that bottom-lines to is ***high-performance high-resolution graphics*** with a wide range of price points.

Let's take a look at what even the low-cost range of this spread can do for your resolution, performance and load-balancing problems.

. . . a rich repertoire of commands, primitives and structurism . . .

Even in the low-cost HP9000/300 you find a feature set with a functional richness that puts you in a ***new graphics domain.***

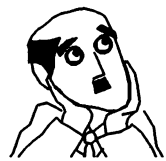
With tools like ***EGS*** (Engineering Graphics System), you can ***zoom*** in on a particular detail and get the positioning you want, right down to the whiskers on the face (actually, a whisker's about 50 microns in diameter and the system has sub-micron resolution capabilities) . . .

Using the ***cell instantiation***, component and level ***display selectivity*** and other functions so essential to CAD applications such as integrated circuit layout, you can build a basic cell one time . . .

. . . and ***scale*** and package it with a given instantiation in a given context . . .

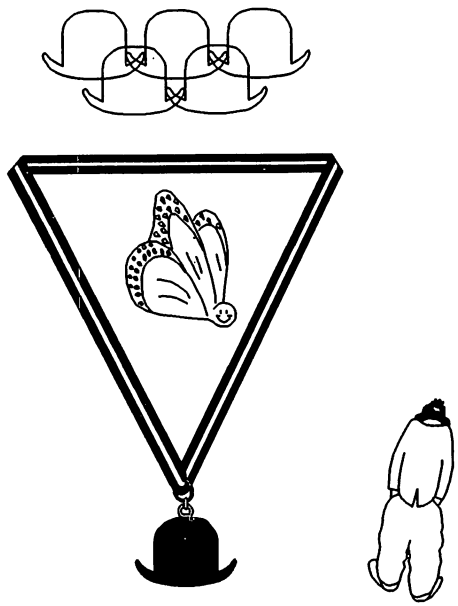
. . . include it in another instantiation with a different context . . .

. . . ***mirror*** the same cell for a different orientation in still another context . . .



. . . build up your ***hierarchical structure*** of cellular components with whatever ***scaling, translation, rotation, mirroring, zooming, panning*** are required, till you have the modules arranged in a multi-level composite that is your complete circuit . . . or whatever it is you're building . . .

. . . you can then take your integrated circuit layout or whatever it is you're building, "***plot to disc***" so you get an ASCII form of the HPGL commands that normally drive a plotter. You then use a ***terminal emulator*** (LAN's on its way, so your 9600 baud can move to the multi-megabit range) to get the vectors from the 9000 to the 3000. Here you've got the full power of TDP, lasers and other technology (such as EGS2FIG in the Contributed Library) to do your final packaging. Once established, your components can go into a library to provide a ***leverage base*** for future fan-out . . .



Epilogue . . .

The odyssey spans computer domains, operating systems, design disciplines. It gives you the rich functionality of the HP9000 CAD/CAM world, with it's high resolution, instantaneous response and natural human interface. It gives you the powerful data-words-graphics capabilities of the HP3000 and the laser printer. And it smoothes your processing load by spreading it out across the appropriate nodes to make your general user population a bit happier as their data processing engine is a little more responsive to their touch.

About the Author ....

Sam Boles *is a Member Technical Staff in the Hewlett-Packard Information Software Operation in Cupertino, California. With HP since 1976, his computer experience started back in the AUTOCODER days of the 1401/1410, migrated thru the 360/370 era, and now focuses on next-generation operating system software. Sam received his MS at UCLA in Information Systems.*

sebiug17 2035 27jan86